A perfectly parallel optimisation for cutback trailing edges

Rob Watson[a] and Paul Tucker.[b]
University of Cambridge, Cambridge, UK, CB2 1PZ

Previous attempts have been made to optimise the performance of film cooling slots for cutback trailing edges, but these have involved the use of steady methods, which have been shown to be inappropriate for accurately capturing the performance of this class of flows. Here, an unsteady method — large eddy simulation on a coarse grid, or very large eddy simulation, (VLES) — is compute the flow. To take advantage of the enormous parallel capacity of modern supercomputers and distributed computing nets, as well as the relatively low cost of VLES, whilst mitigating its lower scope for significant parallelisation, a perfectly parallel evolutionary optimisation process was undertaken. A relatively crude optimisation aim was used to maximise the adiabatic wall film cooling effectiveness averaged over the entire exposed cutback surface, as a proof of concept. The optimising heuristic then used an evolutionary approach to design a turbulator planform, subject to some imposed design restrictions. Six hundred LES type simulations were carried out over 12 generations, and the best performing designs from the last generation is examined. The optimised design showed a considerable improvement in the target metric over the previous experimental geometries. The influence of various geometric parameters on several of the metrics of film cooling is also explored, by mining data from the populations generated over the course of the optimisation. In a targeted optimisation exercise, it is likely that this data could be used to steer the course of the evolution down favourable paths more quickly.

[a] Research Associate, Department of Engineering, rob.watson@eng.cam.ac.uk.
[b] Professor, Department of Engineering, pgt23@cam.ac.uk.

Nomenclature

Δx^+,Δy^+,Δz^+ = non-dimensional wall distance

Δx = finite length scale

Δ = grid spacing

η_{aw} = adiabatic wall film cooling effectiveness

γ = ratio of specific heats

ϕ = convected scalar

ρ = density

τ_{ij}^R = residual stress tensor

C_D = discharge coefficient

C_S = Smagorinsky constant

C_α = Alpha model constant

E_{gen} = generation averaged effectiveness

$\mathcal{F}$ = flux variable

f_{opt}, g_{opt} = optimisation targets

$\dot{m}$ = mass flow rate

$\mathcal{Q}$ = conserved variable

p = static pressure

p_0 = stagnation pressure

t = time

u,v,w = components of velocity

U = fluid speed

I. Introduction

One of the great advantages computational fluid dynamics has over experimental methods is the ease with which automated optimisation exercises can be conducted[1]. In these exercises, target output functions are maximised while compatibility with a given set of input parameters is maintained. An early example of this kind of numerical optimisation was carried out by Hicks and

Henne[2] The argument that these are more easily carried out computationally than experimentally remains true even with the advent of modern technologies such as rapid prototyping and three dimensional printing. One of the most significant reasons for this is how easily the process can be automated — simply set to run, and, without any need for user supervision, producing an improved design[3]. There is also a much greater scope for the parallelisation of computational problems: computational hardware, for all its variety and complexity, will carry out a given set of mathematical instructions in the same way in every instance, and is widely available[4] — something which it is unlikely to be possible to replicate physically.

A variety of approaches for the optimisation of turbine blades have been attempted. Many of these focus on surrogate models, in which local models are built to mimic the design space, with the aim of reducing the local number of calls to the numerical calculation tool. A survey of some of these methods for a variety of applications is given in Jin et al.[5]. A two dimensional study of the use of these methods to design turbine blade profiles was conducted by Peter and Marcelet[6]. Keane[7] extended similar ideas to the design of turbine blades which could cope with geometric uncertainties, but here did not find that any method performed universally better than the others. Simpson et al.[8] compared a kriging-based method to more classical response surfaces as ways of generating the surrogate model, finding the kriging to produce a somewhat better surface for optimising an aerospace nozzle.

Approaches have also been tried which do not depend on surrogacy. Genetic and evolutionary processes mimic, in various ways, the processes associated with natural evolution. Hajela[9] discusses the use of these kinds of approaches, and the benefits they can bring to general complex optimisation problems. Subsequently, Pierret[10] explored the ability of a learning genetic algorithm to optimise a coupled mechanical-flow solver for a fully three dimensional compressor blade design, which was found to perform well. Oyama et al.[11] also used an evolutionary algorithm approach with a fully three dimensional steady solver to realise 19% reductions in entropy generation for transonic compressor blades.

The behaviour of the cutback trailing edge system is dominated by wall-jet type bound von Karmen vortex streets shed from the lip of the cutback blade surface. It has been shown frequently

that steady RANS methods are inappropriate for dealing with this[12, 13].

Unfortunately, the proven alternative, large eddy simulation — and turbulence resolving unsteady simulations generally — have been regarded as unsuitable for use as a design tool, due to their much greater expense when compared with the traditional Reynolds-Averaging techniques. This extra expense is massively compounded when running the many simulations which are required for the successful conduction of an optimisation exercise. It is hoped that it can be shown here that, thanks to a careful problem selection, Moore's law, and strategic fidelity management, optimisation exercises are not necessarily infeasible even when unsteady simulation techniques are needed.

Here, a hypothetical industrial optimisation of a cutback trailing edge is carried out to find the most effective turbulator planform for a fixed cutback blade geometry.

II. The Test Case

The experiments conducted in Karlsruhe by Martini and Schulz were selected as an appropriate test-bed for these optimisations. These cases consist of a single basic geometry, outlined in Figure 1[14]. Inside the cooling cavity, a set of different turbulator layouts were tested. These experiments were also conducted over a range of blowing ratios, M, which is the ratio of momentum between the coolant and the mainstreams:

$$M = \frac{\rho_{cold} U_{cold}}{\rho_{hot} U_{hot}} \tag{1}$$

The turbulator layouts considered in the experiments are also shown in Figure 1.

These experiments represent an engine-realistic Reynolds number, with realistic overall geometries, and a range of internal turbulator layouts, making them a good choice for representing the behaviour of a computational method when considering industrially relevant problems. The basic layout of these experiments was adopted as the universal set-up for this optimisation, with the turbulator pins being rearranged to provide an optimised solution. The boundary conditions which were used throughout the optimisation are also shown in Figure 1. A fixed mass flow rate inflow was chosen for the coolant stream, as this was more susceptible to changes in the passage blockage, and indirectly affecting the blowing ratio.

It had been previously found that the inclusion of turbulent conditions at the inflows made

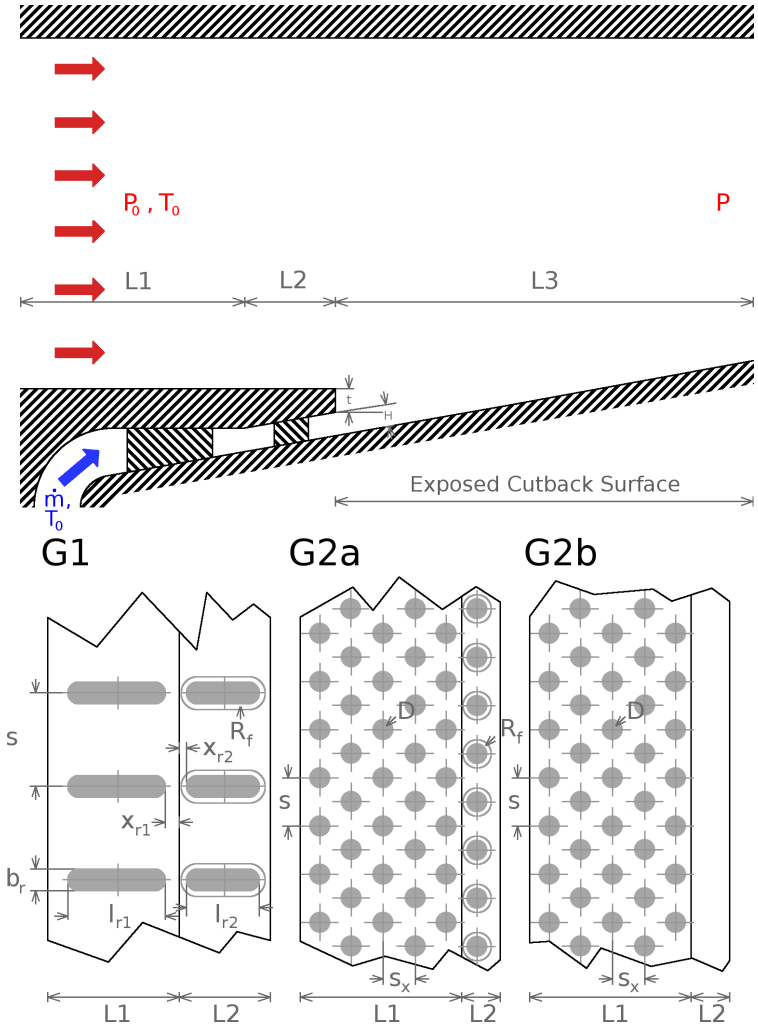

Fig. 1 Karlsruhe case basic geometry, boundary conditions, and experimental turbulator planform [14]

little difference to the film cooling effectiveness results. This is believed to be because of the powerful internal turbulence generation mechanisms, which have a much greater effect on the flow development than any incoming unsteadiness.

The adiabatic wall film cooling effectiveness is a convenient way to measure the performance of a film cooling system. This is given by:

$$\eta_{aw} = 1 - \frac{T_{wall} - T_{cold}}{T_{hot} - T_{cold}} \tag{2}$$

In the experimental data shown here, η_{aw} is often presented as averaged in the spanwise direction, and it is used as a key parameter in the optimisation.

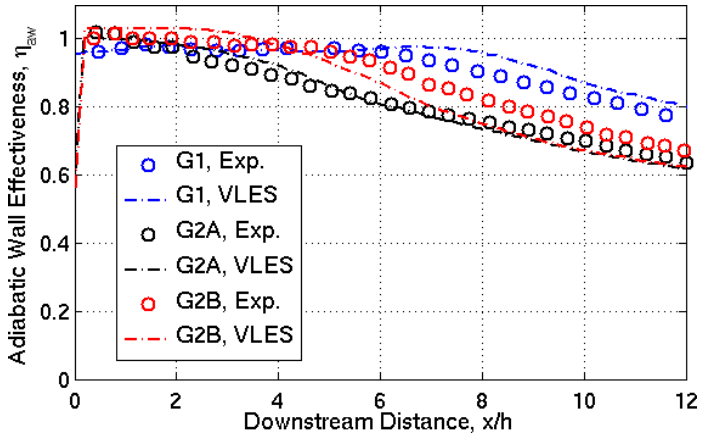

Fig. 2 Comparison of VLES and experimental results for Karlsruhe geometries, M=1.10

III. Numerical Methods

To minimise the cost of this optimisation, a lower cost alternative to fully resolved LES was used, which has been shown to be useful for capturing the behaviour of cutback trailing edge flows. This alternative — large eddy simulation on a coarse grid (VLES) — was found to reduce the simulation cost of each calculation by as much as 90%. Figure 2, reproduced from another paper[13], illustrates that the VLES technique is well able to capture the spanwise averaged film cooling effectiveness performance that was found experimentally.

Very large eddy simulation (VLES) is used here to mean large eddy simulation on a coarse grid — the same LES methodology is applied, but the usual rule-of-thumb of resolving about 90% of the kinetic energy of the flow is substantially relaxed. The principle of its application is that large unsteady structures dominate the mixing behaviour of certain types of flows, with the subgrid model being "good enough" for approximately capturing the behaviour of the significantly less important smaller scales. Thus, VLES is a technique which may be well suited to simulating the behaviour of "top-down" flows, which are dominated by large geometrically generated flow structures, as opposed to "bottom-up" flows, which rely on the interplay of fine turbulent structures, even in the near wall areas. After conducting a large number of simulations, a much coarser mesh which was still capable of capturing the behaviours of the experimental geometries was developed. The meshes used for the VLES calculations thus comprised around 3.5 million nodes, as compared to approximately 20 million for a typical LES mesh. The calculations were then performed in the same way as for

Direction	ND Wall Distance of Mesh
$\triangle x^+$, streamwise	~20
$\triangle y^+$, wall normal	~15
$\triangle z^+$, spanwise	~80

Table 1 The near wall mesh sizing that was used for the very large eddy simulation calculations

the more finely resolved LES solutions. The statistics for the first off wall grid node used in these meshes are given in Table 1, and are far coarser than would normally be used. An unstructured but prismatic grid was used to maximise the ability to vary the mesh shape in the span-normal directions, which was particularly useful given the relaxed boundary layer resolution requirements.

An industrial code, HYDRA, was modified from an approach based on the common second order approximate Riemann solver of Roe[15]. Both the original code and its modified descendant are second order and edge-based, and form dual median control volumes in a node-centred manner. In the modified code, the VLES calculations were carried out using the second order KEP scheme proposed by Jameson[16], and described and tested elsewhere[17]. For completeness, the semi-discretised equations for the scheme are given by:

$$\frac{\partial \mathcal{Q}}{\partial t} + \frac{1}{\Delta x}\left\{\mathcal{F}_{i+\frac{1}{2}} - \mathcal{F}_{i-\frac{1}{2}}\right\} = \frac{\partial \mathcal{Q}}{\partial t} + \frac{1}{\Delta x}(\bar{\rho}_{i+\frac{1}{2}}\bar{u}_{i+\frac{1}{2}}\bar{\phi}_{i_{i+\frac{1}{2}}} - \bar{\rho}_{i-\frac{1}{2}}\bar{u}_{i-\frac{1}{2}}\bar{\phi}_{i-\frac{1}{2}}) = 0 \tag{3}$$

where:

$$\bar{\phi}_{i+\frac{1}{2}} = \frac{1}{2}(\phi_{i+1} + \phi_i) \tag{4}$$

As this is a kinetic energy preserving low dissipation scheme, a subgrid scale turbulence model is needed to capture the influence of scales below that of the implicit grid filter. This is carried out by the formulation of the mixed Alpha model given in Liu et al.[18], by splitting the residual stress tensor,τ_{ij}^R , into linear and non-linear parts:

$$\tau_{ij}^R = \tau_{ij}^L + \tau_{ij}^N \tag{5}$$

The linear term is given by the standard Smagorinsky model[20], and the non-linear term is given by a series of cross-gradient terms:

$$\tau_{ij}^L = -2\rho\left(C_S\Delta\right)^2\left|\overline{S}\right|\overline{S}_{ij} \tag{6}$$

$$\tau_{ij}^N = C_\alpha\rho\Delta^2(\partial_k u_i\partial_j u_k + \partial_k u_i\partial_k u_j + \partial_i u_k\partial_j u_k) \tag{7}$$

The mixed Alpha model is able to replicate more of the physical turbulent behaviours than the classical constant Smagorinsky, such as elements of the reverse energy cascade. It has been suggested that it is competitive in terms of accuracy with dynamic Smagorinsky-Germano formulations[18], but because it does not need a secondary explicit filter to be applied — which is especially expensive with unstructured formulations — is substantially cheaper.

A three-step Runge-Kutta method was used to explicitly integrate the solution in time.

IV. The Parallel Optimisation Process

To discuss how an optimisation process works, it is first necessary to consider a multi-dimensional design space. This consists of the hypothetical set of all possible designs which are compatible with the stipulated constraining specifications — it contains every possible iteration of the various design parameters. Every point in this design space is assigned a value indicating how effectively that point fulfils a certain design criterion. This criterion is known as the test function. Thus, the optimisation space consists of an $n+1$ dimensional scalar field, where n is the number of design parameters. At its most basic, the entire optimisation process is carried out in order to find the deepest trough or highest peak in this test function within its constrained design space.

For illustrative purposes, a two dimensional design space is proposed — the value of some function must be maximised while two independent parameters are allowed to vary. A variety of methods are available for use in an attempt to find the highest peak in this surface.

The most obvious of these methods for finding the maxima of such a function is the basic serial hill climbing approach, in which a starting point is randomly chosen within the design space, and the value of the test function at that point determined. A second test point is then chosen a small distance away in a random direction. Having evaluated the value of the test function at both points, the direction in which a positive gradient is found is selected, and a third test point placed a short distance away in that direction. This process is repeated, and the test location "climbs the hill" towards the maximum value, as seen in Figure 3. Once every possible direction leads to a decrease in value, the maximum is reached, and the search is complete.

One of the most significant problems with this simple approach is immediately apparent — depending on the precise shape of the test function surface, and on the location of the initial seed

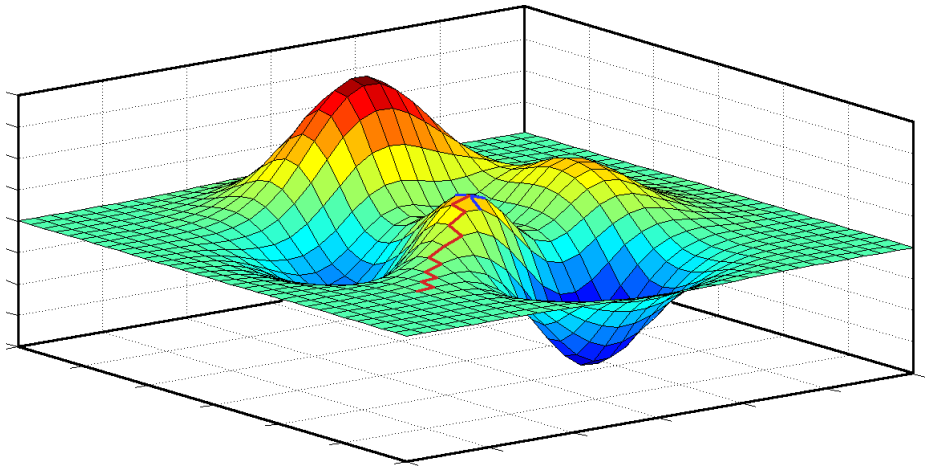

Fig. 3 Hypothetical two dimensional design space, imperfectly solved by a serial hill climb process[21]

point, it is entirely possible for the algorithm to climb the wrong hill: to get trapped in a local maximum, without having found the location of the global maximum. This is in fact the result of the climbing process that has taken place in Figure 3.

A secondary problem with this approach is that it requires test function to be analysed sequentially, one test after another other. In the case of conducting an unsteady simulation, the values of the test functions are very expensive to analyse indeed. If a large number of these had to be carried out one after the other, the real time taken could be very considerable. To accelerate this, optimisation algorithms which allow large numbers of test functions to be analysed simultaneously without reference to each other have been developed. In effect, the analysis of each test function is carried out on an individual group of processors which do not have to communicate with the other groups during the evaluation.

This ability to parallelise is particularly convenient for this case. Modern hardware has tended to increase the number of processing cores, rather than the computational power of individual cores. Figure 4 illustrates the recent trends in average processor core clock speeds among the top500 supercomputers[19]. Despite the continued growth in total computational power, the average clock speed has been almost static for the last decade. To solve the relevant partial differential equations using HYDRA, it is necessary for the various processors working on a single problem to exchange information with each other — the required exchange information is contained within the

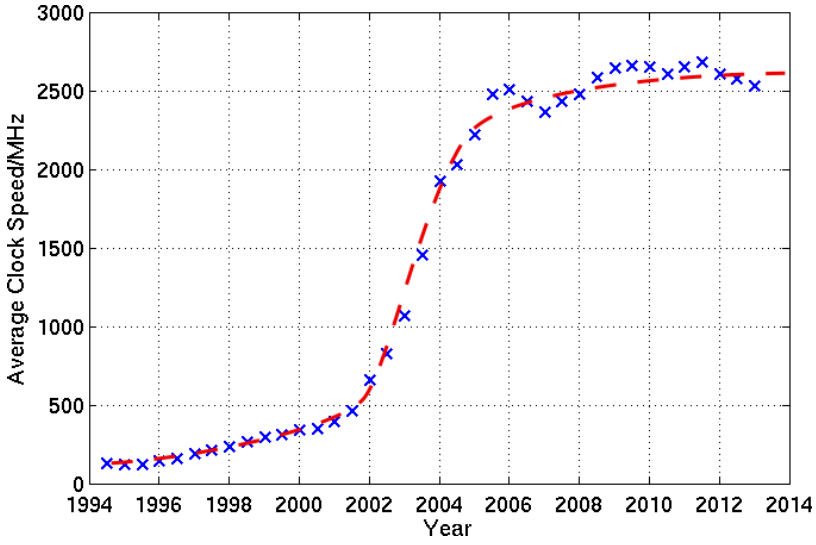

Fig. 4 Historical trends in supercomputer processor core clock speeds

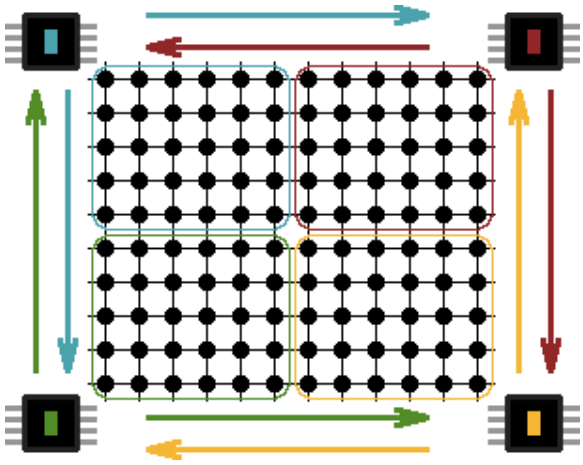

Fig. 5 Message passing and the need for processors to communicate to solve large problems

communication halo. This requirement is shown schematically in Figure 5 - the four processors working on a single large problem must communicate substantially to solve it.

This necessity for communication limits the number of processors which can usefully work together on a problem — eventually, the cost of communicating between more processors will outweigh the computational benefits of adding them, at which point the problem becomes saturated. This implies that there is a lower limit to the time in which modern hardware can solve a single VLES problem — for the present purposes, there is a limit to how quickly the test function at any given point in the design space can be evaluated.

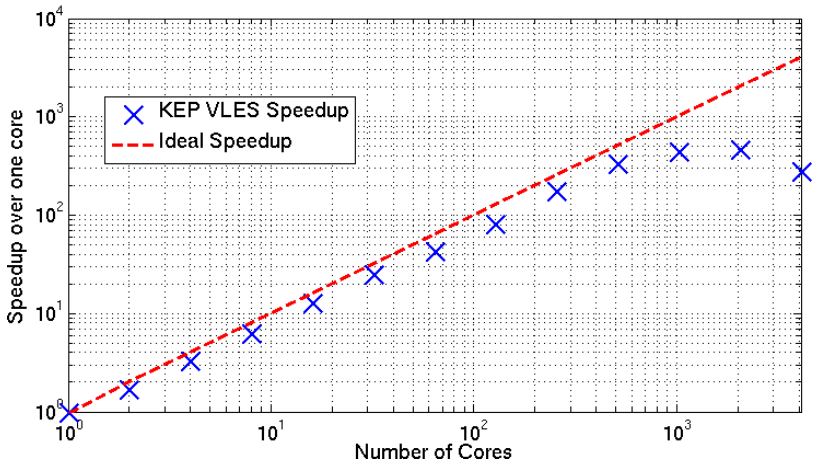

Fig. 6 Ideal scaling vs. strong scaling for KEP VLES on 3.8 million node cutback trailing edge mesh on HECToR

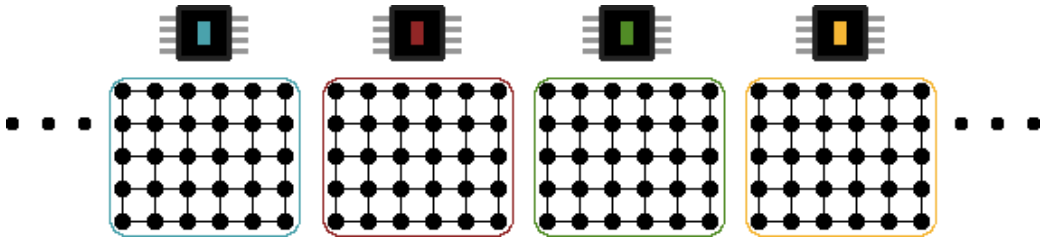

Fig. 7 Embarrassing parallelisation eliminates need for inter-processor communication, and therefore eradicates communications loss

Unstructured solvers such as HYDRA tend to be fairly difficult to use efficiently with extremely parallel hardware, and this problem is compounded by the fact that as mesh size reduces, the amount of communication increases relative to the volume of calculation. As such, it is found that for unstructured VLES, these simulations begin to saturate if more than around 500 cores are used. The effect on calculation speed which the addition of more cores has on HECToR, a UK national supercomputer, is shown in Figure 6.

By making use of the family of optimisation techniques which solve for populations of solutions side-by-side, rather than for single sequential evaluations of the test function, a situation can be achieved which is illustrated schematically in Figure 7, avoiding the restriction of saturation, and allowing optimisations to be carried out in feasible real time. In effect, Figure 7 improves on Figure 5 by solving many smaller problems simultaneously, eradicating any communication loss.

Of course, there is no particular reason why problems of this type need to be run on a supercomputer — when only a few processors are working on each problem, the speed of the interconnects is not hugely relevant. When it is considered that, as an approximate figure, less than 1% of the total general purpose computing power in the world is contained within the sum of all of the top500 systems, the potential power of distributed computing systems to solve highly parallel problems quickly becomes apparent. This could avoid large capital investments in supercomputing resources, albeit at the expense of running costs. A quick calculation suggests that the electrical power costs of running a single generation of this optimisation on a distributed computing network of 200W per core would be around twice that of the proportion of running costs on a supercomputer.

These population-based heuristics come in many varieties — such as particle swarm, firefly, and memetic and genetic algorithms. For simplicity, we here make use of the evolutionary genetic approach.

V. Genetic Algorithms

The genetic approach to optimisation aims to mimic the behaviour of natural selection over a number of generations by promoting the survival of favourable traits. This is achieved by calculating a test function for each design. Those individuals with superior test functions are allowed to pass their traits on to the next generation, where they are recombined with those of other survivors. Thus, fitness to survive is enforced by the prevention of poorly performing individuals from reproducing. Over a number of generations, this leads to improved designs.

In order to carry this out, a stream of data which is capable of reproducing any point in the design space is generated. The precise values of this data stream fix the details of a design. This mimics the genetic code of an organism in biological natural selection. It is these data streams which are passed and recombined between generations to ensure the survival of positive design traits through the optimisation. A schematic recombination of two of these data streams from two survivors into a new offspring design is shown in Figure 8.

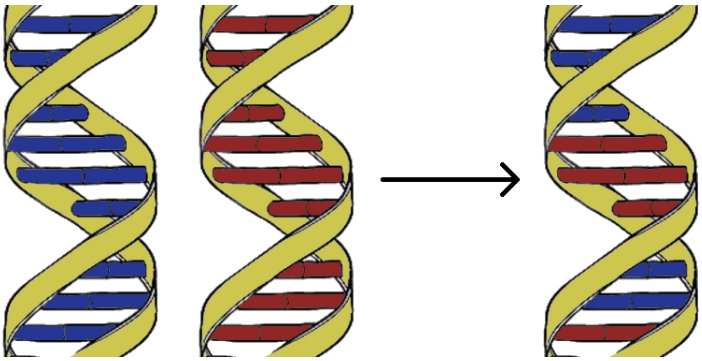

Fig. 8 Genetic recombination to produce an offspring from two surviving parents

Parameter	Role
n_x	Number of pin rows
l_x(n_x)	Length of each pin row
s_y(n_x)	Spanwise pitch of each pin row
r(n_x)	Width and radius of each pin row
o_y(n_x)	Spanwise separation of each pin row
o_x(n_x)	Streamwise offset of each pin row

Table 2 Parametrisation of the turbulator planforms

VI. Parametrisation, Inviability and Success Metrics

A. Parametrisation

The first step in carrying out a design optimisation is to divide the design space into a set of parameters which, between them, carry the necessary design information. For the purposes of this optimisation, the parameters contain the turbulator pin layout data. Figure 9 shows an arbitrary pin layout, and its parametrisation. The parameters contained in each plan file are listed in Table 2. A complete listing of all the values of these parameters for a single design would, in the lexicon of genetic processes, be its genome.

Making use of such parametrisation allows rapid and automated generation of corresponding CFD meshes from the universal geometry, and provides a quick way to vary the turbulator layout between generations of the optimisation.

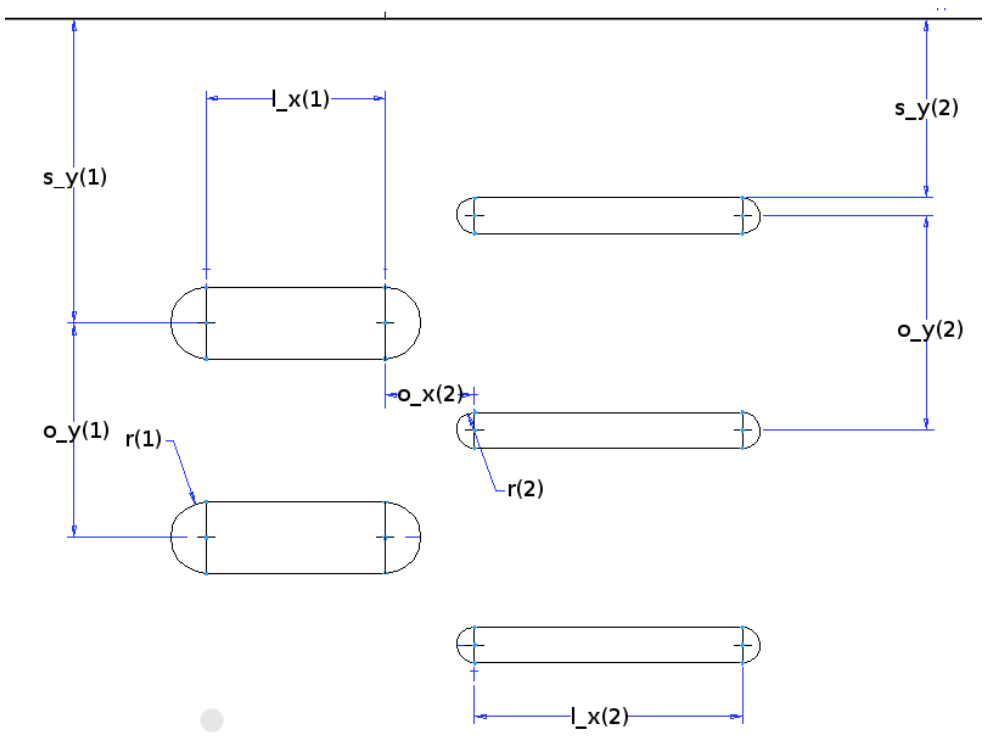

Fig. 9 Parametrisation of the turbulator planforms

B. Inviability

Inevitably, some pin layouts will violate desirable or required design constraints. For example, it is likely that the best performing slot design would be one that was completely empty of pins. Of course, this is a trivial result — in a real engine, the turbulators are necessary for both promoting heat transfer and guaranteeing blade structure. To avoid this, a lower limit on the passage blockage has been assigned - requiring no less than 10% of the passage to be blocked. Equally, an upper limit of 25% passage blockage is applied. Again, to ensure an even distribution of turbulators, for internal heat transfer and structure, the blockage up and downstream of the cavity midsection must be within 30%. Designs which violate these constraints are immediately rejected as inviable designs, and are regenerated until one is determined which satisfies them.

C. Success Metrics

As discussed, any optimisation method requires a target function — some technique for quantifying the relative performance of different designs. Here, our given performance metric is the

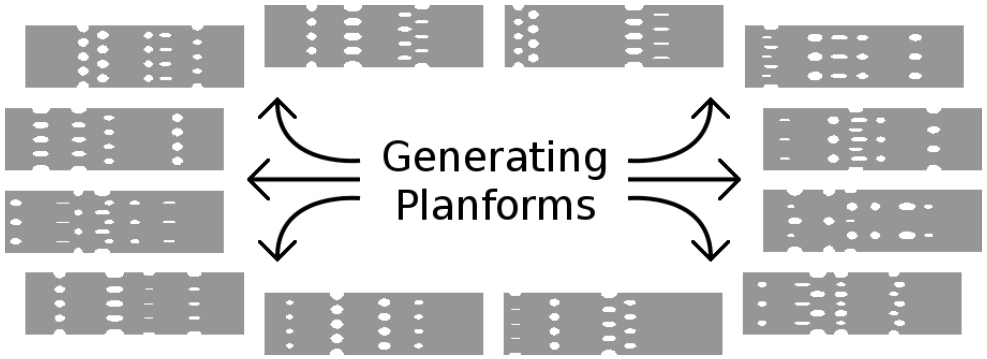

Fig. 10 Set of randomly generated turbulator planforms for the initial population

average value of the adiabatic wall film cooling effectiveness across the entire exposed blade surface.

$$g_{opt} = \frac{1}{A} \int_0^{L_Y} \int_0^{L_X} \eta_{aw}(x, y)\, dx\, dy \tag{8}$$

This approach has the advantage of easily quantifying the performance of the geometry as a single easily understood number. More complicated measures could be constructed for more specific tasks — for example, a weighting which gives a bonus to a more even lateral spread of cooling effectiveness, to favour designs which sharp thermal gradients over the exposed blade surface.

However, this method gives a clear and unambiguous number, which is ideal for the illustrative purposes of the benefits of absorbing unsteady methods into the wider design process.

D. Evolutionary Methods

Having parametrised the design space, stochastic initialisation of a progenitor population could be carried out. Effectively, this means that individual members of the first generation were assigned their genes at random, which survived if these genes resulted in a design which corresponded to the applied contraints. Some of the turbulator planforms which were generated in this way are shown in Figure 10, which shows the range of more exotic planforms which could be generated using this parametrisation scheme and given constraints on viability. Sixty individuals were generated in this way for the initial population.

The performance of each planform was then evaluated by conducting a VLES simulation, storing the time average, and returning the value of g_{opt}. Each VLES calculation took around 2000 core hours on AMD Opteron *Interlagos* 2.3GHz processors. The performance of each individual within

the population was then ranked by the test function, g_{opt}. With this ranking, the evolutionary stage of the optimisation proceeded by culling the worst performing 40% of the individuals. Of the 36 surviving population members, each was assigned a "probability to reproduce" based on their place in the performance hierarchy. The best performing geometries were twice as likely to reproduce as the worst performing which had survived the cull, with a linear probability distribution between them. Sixty pairs of parents were then generated based on this probability distribution to produce the next generation of children.

The child received its parameters at random from each parent, as illustrated in Figure 8. Each parent's gene had a $\frac{1}{2}$ chance of being represented in the child's genome.

As well as the transmission of genes, the genetic algorithm also allows for some copying errors, as occur in biological DNA replication processes. In terms of the optimisation problem, this leads to more exploration of the design space not captured by the genes present in the first generation. The three copying errors which were allowed were: *gene drift*, in which, two percent of the time, a normally distributed probability of variation around the parental value, with a standard deviation of 5% of the starting value was applied; *gene transposition*, in which the values of two consecutive genes in the child's genome are accidentally swapped, which was allowed to happen in 0.1% of cases; and *complete mutation*, in which a gene completely failed to copy from either parent, and a randomly value was instead generated, which was allowed to happen in 0.01% of copying tasks. A completed offspring breeding task is shown in Figure 11. The influence in the child design is apparent — for example, the length, position, and aspect ratio of the third pin row is quite common to both and is faithfully reproduced, while the fourth pin row shows much of the shape of the left parent with the positioning of the right. It also contains some novel features, introduced through the mutation process. If a child was generated which violated one of the viability constraints, a second child was re-bred from the same parents, and so on, until a viable child was produced. The mutation and culling rate coefficients were chosen by making use of values which had worked well on the optimisation of much simpler and cheaper problems.

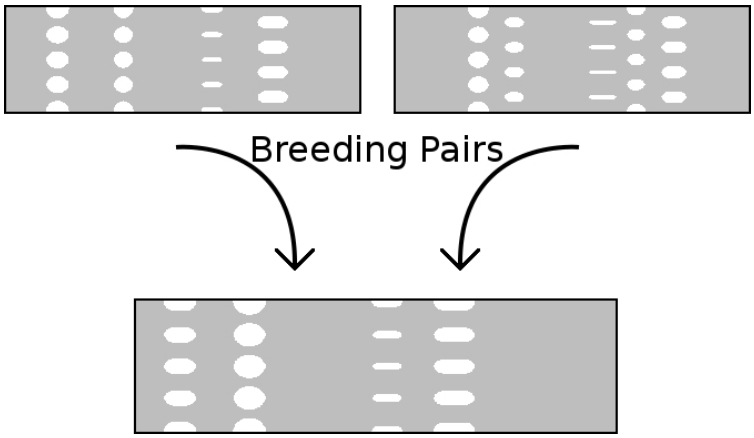

Fig. 11 Breeding a new design from two surviving parents

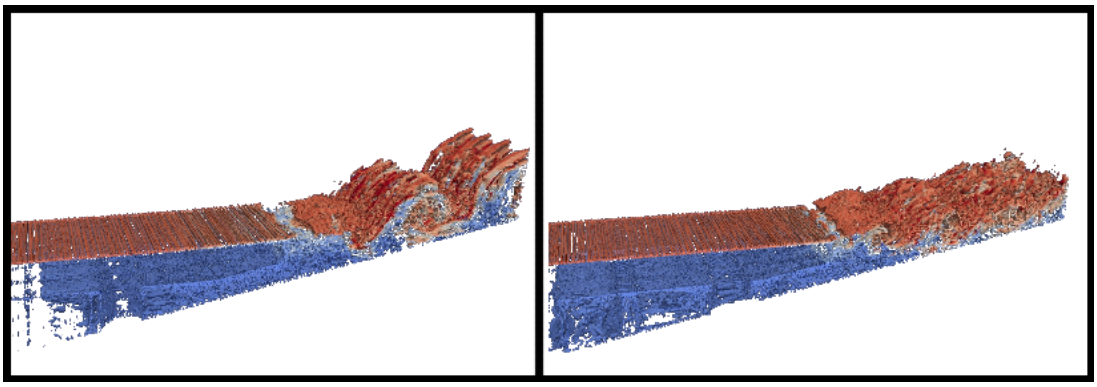

Fig. 12 Comparison of two flow patterns. Instantaneous isosurfaces of Q-criterion, coloured by static temperature

VII. Results

A. The Progress of Evolution

Interactions between the coolant and main streams can generate a wide variety of flow structures and behaviours, despite the apparently small changes to the turbulator layout. Figure 12 shows just two instantaneous Q-criterion profiles of two of these flows, to highlight the difference in behaviour that the turbulator layout can produce. Clearly, the left hand flow is producing significantly larger and more stratified rollers than the right. The difference in the length scales of the unsteadiness which these plots show is invariant with time.

Having advanced the evolutionary process through several generations, it is possible to see how the behaviour of the population as a whole has changed. This was achieved by averaging the

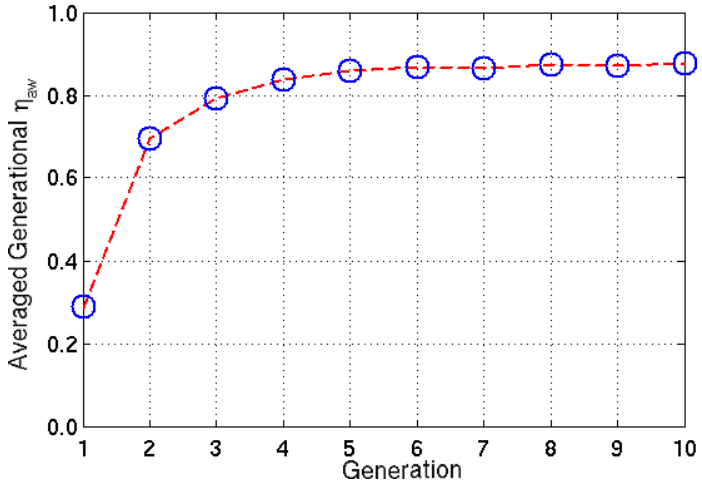

Fig. 13 Improvement in mean performance of each generation through the optimisation

parameter of interest over the entire population of each generation:

$$E_{gen} = \frac{\sum_{i=1}^{\#(gen)} \frac{1}{A} \int_0^{L_Y} \int_0^{L_X} \eta_{aw}(x,y)\, dx\, dy}{\#(gen)} \tag{9}$$

Figure 13 shows how this average performance increases as the generations are advanced. As the optimisation was started from a fairly poor initial guess (the randomly generated initial population), generation one performs extremely poorly, and there is a rapid improvement as the worst designs are rapidly culled. After about the sixth generation, the average rate of improvement tails off.

This tailing off is partly due to the fact that the designs are approaching the optimum, and partly because the rate of mutation is relatively low, so the genes which result in a poorly performing geometries have been weeded out, but the nearby design space is only explored relatively slowly through mutated genes.

B. Empirical Relations

As the best and worst performing members of each generation have converged, the performance of various gene expressions can be considered — giving, effectively, empirical relations between design parameters and performance. The relationships shown between the integrated film cooling effectiveness and the parameter of interest have been taken from the final three generations, when the gene pool has become largely saturated with "good" genes.

Figure 14 shows the development of the relationship between the adiabatic wall film cooling effectiveness and the total void fraction of the coolant cavity through generations. The red represents

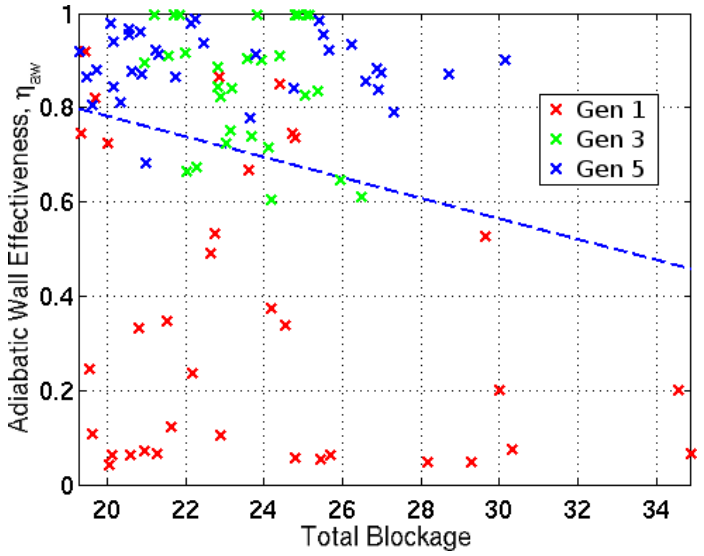

Fig. 14 Generational development of film cooling effectiveness vs. total blockage

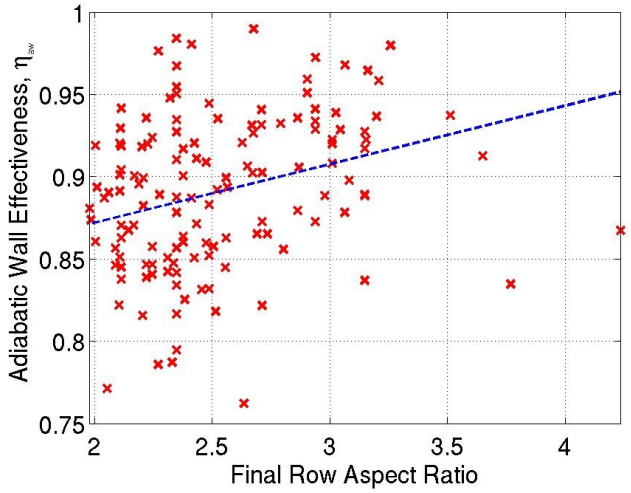

Fig. 15 Correlation between the film cooling effectiveness and final pin row aspect ratio

the first, stochastically generated population. The green is the third generation, and the blue the fifth. Here, the performance gains over the generations are clear, particularly between the red and the green. It is also apparent that the total blockage — as a metric correlated with film cooling effectiveness — is reduced over the optimisation, although the fifth generation (blue) are clustered around two values, one of around 21 and one of around 26 percent void fraction.

Figure 15 shows the correlation between the aspect ratio of the pins in the most downstream pin row and the averaged film cooling effectiveness, the gradient of the trend line of which indicates

	$\bar{\eta}_{aw}$	C_D	x_{core}	T_{max}
Final row aspect ratio	1.0000	0.3271	0.8339	0.1527
Final row blockage ratio	0.8675	0.4991	1.0000	0.3345
Total void fraction	0.7775	1.0000	0.8816	1.0000
Average pin radius	0.6325	0.2781	0.4735	0.2410
Downstream void	0.6138	0.3999	0.9659	0.4325
Average aspect ratio	0.4525	0.4370	0.3973	0.1630
Upstream void	0.3513	0.4174	0.2548	0.1561
First moment of area	0.1850	0.1052	0.0174	0.2245
Final pin row spacing	0.1600	0.5243	0.3562	0.4228
Upstream pin row x	0.1438	0.2785	0.1463	0.3470
Average fin length	0.1375	0.2900	0.0101	0.0889
Middle void	0.1263	0.3181	0.0398	0.5453
Downstream pin row x	0.1025	0.3837	0.2795	0.0991

Table 3 The normalised slope of different geometrical parameters against potential design performance metrics over optimisation

that this is the most important parameter. This suggests that longer, more slender pins should be used in the final row.

The correlations are, of course, weak — there is no requirement for the genetic algorithm to alter only one variable at once. The correlations which seem to have the most influence tend to be with geometric parameters that are likely to distort the two dimensional flow in the near lip part of the coolant cavity.

Table 3 shows how various geometrical parameters influence the behaviour of various performance metrics — the averaged adiabatic wall film cooling effectiveness, the discharge coefficient, the coolant core length, and the maximum wall temperature. This table is compiled from the slope of Theil-Sen estimator lines, such as that seen in Figure 15. These are first normalised by the geometric range exhibited in the optimisation, and then ranked, ultimately being divided by the highest ranked value. In this way, an approximate measure of the relative importance of each of the geometrical parameters is produced.

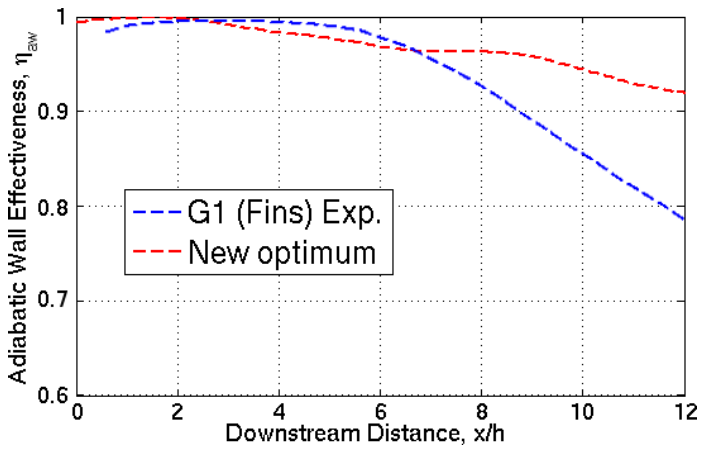

Fig. 16 Comparison of new optimum design and the best performing experimental case inline fins

This lets us see that according to our correlations, the aspect ratio of the pins in the most downstream turbulator row is of vital importance to the average film cooling effectiveness over the entire exposed cutback surface. This corresponds to the aerodynamic idea that the less disturbance there is to a two dimensional jet, the better the film cooling coverage will be. This is supported by the second most important parameter to film cooling effectiveness — the fraction of the passage blockage caused by the final row. The actual streamwise positions of the pin rows are found to be relatively unimportant to the film cooling effectiveness measurements, within the constraints of the optimisation.

Across all of the performance measures, it is the void fraction of the planform, the volume of space in the coolant cavity divided by the volume of space in an empty cavity, which is found to be dominant, particularly for the discharge coefficient. Again, this is intuitively reasonable.

The rules of thumb, then, for the design of these systems should be to minimise the disruption to the two dimensional nature of the coolant jet, and to minimise the blockage in the coolant cavity. Beyond this, it is the use of optimisation combined with accurate solution methods which will give the gains in engine performance which are required to remain competitive.

C. Final Design

The time averaged adiabatic film cooling effectiveness of the best performing turbulator planform from the final generation is shown in Figure 16. The design of the planform is not shown.

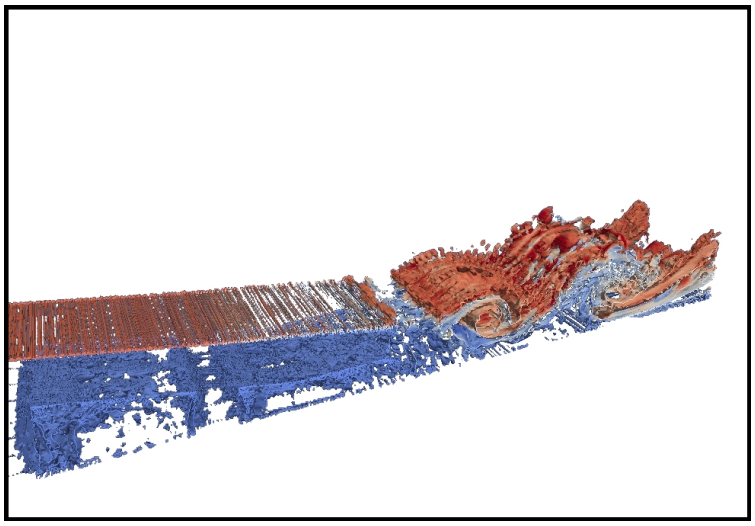

Fig. 17 Optimised unsteady flow field, Instantaneous isosurfaces of Q-criterion, coloured by static temperature

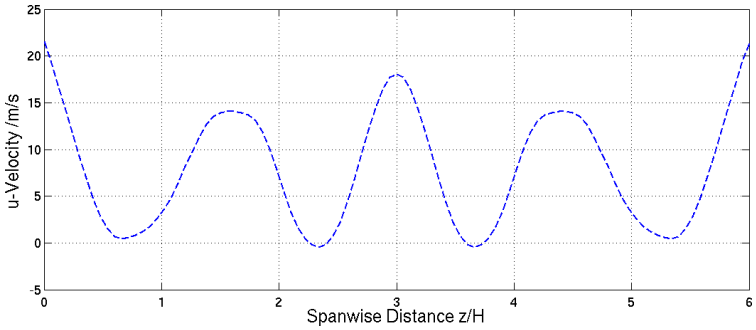

Fig. 18 Time averaged streamwise velocity profiles in spanwise direction at coolant cavity exit

It is evident that the new design performs significantly better than the best performing of the Martini and Schulz experimental results — the inline fins — at an equal blowing ratio, although the new design does tend to drop more smoothly than the fin layout design, leaving it slightly worse between around four and six slot heights downstream.

The flow which this turbulator design produces can be seen as an unsteady snapshot in Figure 17.

Figure 19 shows the adiabatic wall temperature over the exposed cutback surface at the same instant that the flow visualisation in Figure 17 was taken. Interestingly, there is not a spike in wall temperature behind each pin — instead, it appears that the coolant jets have coalesced. This slight tendency towards jet coalescence can be seen in Figure 18, which plots the mean streamwise

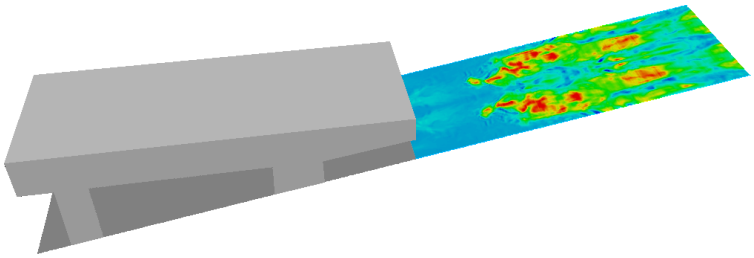

Fig. 19 Optimised unsteady flow field, instantaneous countours of adiabatic wall temperature over cutback surface

velocity profiles at the pin mid-height at the exit of the coolant cavity. This feature was stable over the timescales used in these calculations. Experimentally, it has been found that at lower blowing ratios, there can exist a number of metastable regrouping patterns which shift over time[12]. In real engines, the existance of high thermal gradients over the blade surface may be unacceptable, and the phenomenon of jet coalesence must be taken into account to avoid and ameliorate such difficulties.

Of course, with such a deliberately naive optimisation target and loose restrictions on the viable flows, it is unlikely that this optimisation will truly generate a useful planform. The imposed selection pressure to maximise averaged film cooling effectiveness has not directly led to a good candidate for an engine film cooling system. Instead, a more nuanced trade-off between the various parameters is required to give a directly relevant solution.

Crucially, the discharge coefficient, C_D, measures the ratio of ideal mass flow to true mass flow — effectively a measure of the pressure difference across the coolant cavity which is required to force the required mass flow rate for a given blowing ratio through the turbulator array. This measure is given by:

$$C_D = \frac{\dot{m}_{c,real}}{\dot{m}_{c,ideal}} = \frac{\dot{m}_{c,real}}{p_{01}\left(\frac{p_2}{p_{01}}\right)^{\frac{\gamma+1}{2\gamma}} A_{slot}\sqrt{\frac{2\gamma}{(\gamma-1)RT_{01}}\left[\left(\frac{p_{01}}{p_2}\right)^{\frac{\gamma-1}{\gamma}}-1\right]}} \tag{10}$$

With the current optimisation set up, there is no limit on how low this can go — effectively, no limit to how much work must be done by the flow to force itself through the fins. This is obviously undesirable in a real engine set up, where this pressure ratio will largely determine the mass flow rate, and thus the blowing ratio.

The geometry which has been optimised to maximise averaged film cooling effectiveness has a rather low discharge coefficient. For comparison, those tested by Martini and Schulz have values of approximately 0.6 for the inline fins case, and about 0.45 for the staggered pins at the same blowing ratio. The film cooling coverage geometry suffers here, which makes it unsuitable for genuine engine applications, with a discharge coefficient of only 0.18. A more practical optimisation method would ensure that the two functions were maximised together, possibly by maximising the value of some combined function, such as:

$$f_{opt} = C_D \times \frac{1}{A} \int_0^{L_Y} \int_0^{L_X} \overline{\eta}_{aw}(x, y)\, dx\, dy \tag{11}$$

An alternative approach would be to place a minimum value on the acceptable discharge coefficient, whilst still maximising the film cooling effectiveness performance alone. This could be relatively straightforwardly achieved by using the discharge coefficient calculated from the precursor RANS simulations. Although very poor for predicting film cooling performance, RANS is adequate for getting a reasonable estimate of the resulting discharge coefficient. Any geometries which fell below the specified threshold could then be regenerated before the much more costly VLES simulation is carried out.

One of the powers of the population based methods is that the search algorithm has considered a wide range of the solution space. This means that by searching through the population, the maximum value for f_{opt} which has been previously encountered by the search can be found. By doing this, there is a reasonable chance of an adequate solution, despite the progress of the optimisation having been targeted by a different metric. By doing this, a reasonably well performing film cooling planform is found, with a much more respectable discharge coefficient of 0.38. The time averaged film cooling performance of this flow compared to that of the best performing experimental data in shown in Figure 20, again, the planform itself is not shown.

The best performing f_{opt} geometry has a somewhat higher total void fraction within the coolant cavity, which is very strongly correlated with discharge coefficient as an effective measure of wetted surface area. It is rather less strongly correlated with averaged film cooling effectiveness.

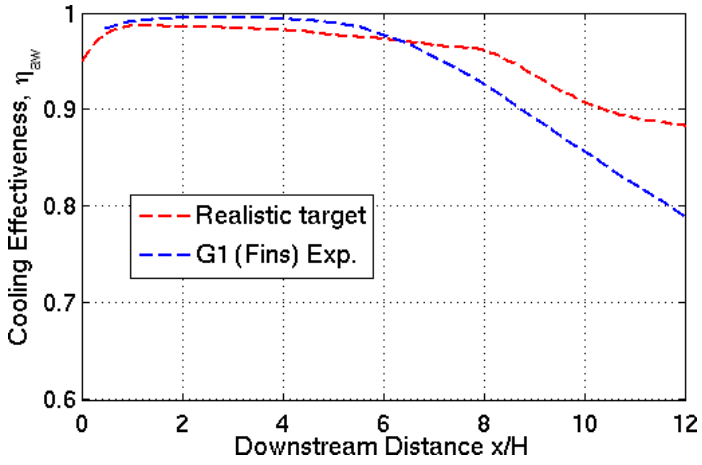

Fig. 20 Comparison of f_{opt} design and original inline fins

VIII. Conclusions

By reducing the mesh size, the scope for parallelising each individual calculation was reduced, but this can be reclaimed by adopting population-based heuristics to conduct design iterations. During the simulation, up to 15000 cores were being deployed at once, with very little communications loss, and without saturating the problem. This approach also leaves considerable scope for the problem to be extended further — there were only 60 individuals in each population, which could easily be expanded to take advantage of increases in available computational width, and to enjoy the fruits of the extra exploration of the design space.

By focusing on purely maximising the average film cooling effectiveness, an excellent performer for this metric has been generated, again confirming the ability of the genetic algorithm to optimise problems. The resulting solution proved to be a significantly better performer by this metric than the best performing "representative geometry" used in the experiments of Martini and Schulz.

However, this optimisation was not concentrating on producing the "best film cooling system". Thought needs to be given to the definition of the test function, in conjunction with the holistic design of the rest of the engine systems to ensure that the performance of the film cooling system is helping the overall efficiency.

It has also been confirmed that for some situations — where traditional steady methods are unable to correctly capture the behaviour of the flows, or even their trends — VLES is a viable alternative, even for optimisation problems, which have previously been viewed as beyond the reach

of turbulence resolving methods.

Acknowledgements

This work was supported by an iCASE studentship from the Engineering and Physical Sciences Research Council, via Rolls-Royce plc. The funding from both organisations is gratefully acknowledged.

This work made use of the facilities of HECToR, the UK's national high-performance computing service, which is provided by UoE HPCx Ltd at the University of Edinburgh, Cray Inc and NAG Ltd, and funded by the Office of Science and Technology through EPSRC's High End Computing Programme.

References

[1] Kipouros, T., Jaeggi, D., Dawes, B., Parks, G., and Savill, M., "Multi-objective optimisation of turbomachinery blades using tabu search," *Evolutionary Multi-Criterion Optimization*, 2005, pp. 897–910.

[2] Hicks, R. M. and Henne, P. A., "Wing Design by Numerical Optimization", *Journal of Aircraft*, Vol 15, No. 7, 1978, pp. 407–412.

[3] Shahpur, S. and Lapworth, L., "PADRAM: Parametric Design and Rapid Meshing System for Turbomachinery Optimisation," *ASME Turbo Expo 2003*, 16th–19th Jun. 2003, Atlanta, Georgia, USA, GT2003-38698.

[4] Harvey, S., "The Design Optimisation of Turbomachinery Blade Rows," *PhD Thesis*, University of Cambridge, 2002.

[5] Jin, R., Chen, W., and Simpson, T. W., "Comparative studies of metamodelling techniques under multiple modelling criteria," *Structural and Multidisciplinary Optimization*, Vol 23, No. 1, 2001, pp. 1–13.

[6] Peter, J. and Marcelet, M., "Comparison of surrogate models for turbomachinery design," *WSEAS Transactions on Fluid Mechanics*, Vol 3, No. 1, 2008, pp. 10–17.

[7] Keane, A. J., "Comparison of several optimization strategies for robust turbine blade design," *Journal of Propulsion and Power*, Vol 25, No. 5, 2009, pp. 1092–1099.

[8] Simpson, T. W., Mauery, T. M., Korte, J. J., and Mistree, F., "Kriging Models for Global Approximation in Simulation-Based Multidisciplinary Design Optimization," *AIAA Journal*, Vol 39, No. 12, 2001, pp. 2233–2241.

[9] Hajela, P., "Genetic search — an approach to the nonconvex optimization problem," *AIAA Journal*, Vol 28, No. 7, 1990, pp. 1205–1210.

[10] Pierret, S., "Multi-objective and multi-disciplinary optimization of three-dimensional turbomachinery blades," *Proceedings of the 6th World Congresses of Structural and Multidisciplinary Optimization*, 30th May–3rd June 2005, Rio de Janeiro, Brazil.

[11] Oyama, A., Liou, M.-S., and Obayashi, S., "Transonic axial-flow blade optimization: Evolutionary algorithms/three-dimensional Navier-Stokes solver," *Journal of Propulsion and Power*, Vol 20, No. 4, 2004, pp. 612–619.

[12] Martini, P., Schulz, A., Whitney, C. F., and Lutum, E., "Experimental and numerical investigation of trailing edge film cooling downstream of a slot with internal rib arrays," *Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy*, Vol 217, 2003, pp. 393–401.

[13] Watson, R. and Tucker, P., "Unsteady simulation strategies for trailing edge ejection," *AIAA SciTech 2014*, 13th–17th Jan. 2014, National Harbor, Maryland, USA, AIAA-2014-1425.

[14] Martini, P., Schulz, A., and Bauer, H.-J., "Film cooling effectiveness and heat transfer on the trailing edge cutback of gas turbine airfoils with various internal cooling designs," *ASME Journal of Turbomachinery*, Vol 128, No. 1, 2006, pp. 196–205.

[15] Roe, P. L. "Approximate Riemann solvers, parameter vectors, and difference schemes," *Journal of Computational Physics*, Vol 43, No. 2, 1981, pp. 357–372.

[16] Jameson, A., "Formulation of kinetic energy preserving conservative schemes for gas dynamics and direct numerical simulation of one-dimensional viscous compressible flow in a shock tube using entropy and kinetic energy preserving schemes," *Journal of Scientific Computing*, Vol 34, 2008, pp. 188–208.

[17] Watson, R., Tucker, P., Wang, Z.-N., and Yuan, X. "Modifying an unstructured Roe solver for large eddy simulation," *AIAA SciTech 2014*, 13th–17th Jan. 2014, National Harbor, Maryland, USA, AIAA-2014-1441.

[18] Liu, Y., Tucker, P. G., and Kerr, R. M., "Linear and nonlinear model large-eddy simulations of a plane jet," *Computers and Fluids*, Vol 37, No. 4, 2008, pp. 439–449.

[19] Strohmaier, E., Dongarra, J., Simon, H., and Meuer, M. "TOP500 supercomputing sites," *http://www.top500.org/lists/2014/11*, November 2014.

[20] Smagorinsky, J. "General circulation experiments with the primitive equations: I. The basic experiment," *Monthly Weather Review*, Vol 91, No. 3, 1963, pp. 99–164.

[21] MathWorks, "MATLAB Primer R2013b," The MathWorks Inc., September 2013.